

What Is Claimed Is:

1. In a system including first and second distinct computers respectively having first and second different operating systems, a cross-platform process scheduling method comprising the step of:

5 scheduling a first process compatible with the first operating system and a second process compatible with the second operating system to respectively execute on the first and second computers.

10 2. The method of claim 1, wherein a master schedule includes a first process identifier identifying the first process and a second process identifier identifying the second process, the first and second process identifiers being linked together to define an executing sequence of the first and second processes, the method further comprising the step of scheduling the first and second processes to execute on the respective first and second computers according to the defined executing sequence.

15 3. The method of claim 2, wherein the master schedule includes one or more conditional inter-relationships between the first and second processes, the method further comprising the step of scheduling the first and second processes to execute based on the one or more conditional inter-relationships.

20 4. The method of claim 3, wherein the one or more conditional inter-relationships include a success criteria associated with the first process, the method further comprising the steps of:

executing the first process;

comparing the success criteria to execution results produced by the first process; and

25 determining whether the first process executed successfully based on the comparison step.

5. The method of claim 4, wherein the master table includes a third process identifier identifying a third process, the method further comprising the alternative steps of:

5 executing the second process when the first process executed successfully according to the determining step, and

 executing the third process but not the second process when the first process did not execute successfully according to the determining step.

6. The method of claim 5, further comprising the steps of:

10 sending a command to the first computer to initiate execution of the first process on the first computer;

 receiving a result message from the first computer, the result message including the execution results produced by the first process; and

15 sending a command to the second computer when the first process executed successfully according to the determining step to initiate execution of the second process on the second computer.

7. The method of claim 2, further comprising the step of monitoring processor loading associated with the first and second computers and adjusting the executing sequence based on the processor loading.

20 8. The method of claim 2, wherein the master table includes a priority associated with each process identifier, the method further comprising the step of adjusting the executing sequence based on the respective priorities associated with the first and second processes.

9. The method of claim 1, further comprising prior to the scheduling step, the steps of:

25 receiving a first message from the first computer indicating that the first process needs to be scheduled for execution on the first computer;

receiving a second message from the second computer indicating that the second process needs to be scheduled for execution on the second computer; and generating the master schedule based on the first and second messages.

10. The method of claim 9, wherein the generating step includes the steps of:

receiving one or more commands indicating an executing sequence of the first and second processes; and

linking the first and second processes together according to the commands.

11. A system for scheduling the execution of cross-platform computer processes on a plurality of client computers, the plurality of client computers including first and second distinct computers having respective first and second different operating systems, comprising:

a process scheduling computer coupled to the first and second computers, the scheduling computer including a scheduler that schedules a first process compatible with the first operating system and a second process compatible with the second operating system to respectively execute on the first and second client computers.

12. The system of claim 11, further comprising a master schedule that is accessible to the scheduler, the master schedule including a first process identifier identifying the first process and a second process identifier identifying the second process, the first and second process identifiers being linked together to define an executing sequence of the first and second processes, wherein the scheduler schedules the first and second processes to execute on the respective first and second computers according to the defined executing sequence.

13. The system of claim 12, wherein the master schedule includes one or more conditional inter-relationships between the first and second processes, and wherein the scheduler schedules the first and second processes to execute based on the one or more conditional inter-relationships.

5 14. The system of claim 13, wherein the one or more conditional inter-relationships include a success criteria associated with the first process, and wherein the scheduler includes:

means for executing the first process;

10 means for comparing the success criteria to execution results produced by the first process; and

means for determining whether the first process executed successfully based on a comparison result produced by the comparing means.

15 15. The system of claim 14, wherein the master table includes a third process identifier identifying a third process, and wherein the scheduler includes:

means for executing the second process when the first process executed successfully according to the determining step; and

means for executing the third process but not the second process when the first process did not execute successfully according to the determining step.

20 16. The system of claim 15, wherein the scheduler further comprises:
means for sending a command to the first computer to initiate execution of the first process on the first computer;

means for receiving a result message from the first computer, the result message including the execution results produced by the first process; and

25 means for sending a command to the second computer to initiate execution of the second process on the second computer when the determining means determines the first process executed successfully.

17. The system of claim 12, wherein the scheduler comprises means for monitoring a processor loading associated with the first and second computers and adjusting the executing sequence based on the processor loading.

5 18. The system of claim 12, wherein the master table includes a priority associated with each process identifier, and wherein the scheduler includes means for adjusting the executing sequence based on the respective priorities associated with the first and second processes.

10 19. The system of claim 11, wherein the scheduler comprises:
means for receiving a first message from the first computer indicating that the first process needs to be scheduled for execution on the first computers;
means for receiving a second message from the second computer indicating that the second process needs to be scheduled for execution on the second computer; and
15 means for generating the master schedule based on the first and second messages.

20 20. The system of claim 19, wherein the generating means includes:
means for receiving one or more commands indicating an executing sequence of the first and second processes; and
means for linking the first and second processes together according to the commands.

25 21. In a system including first and second distinct computers respectively having first and second different operating systems, a computer program product comprising computer usable media having computer readable program code means embodied in the media for causing application programs to execute on a computer processor to perform cross-platform computer process scheduling, the computer readable program code means comprising:

first computer readable program code means for causing the processor to schedule a first process compatible with the first operating system and a second process compatible with the second operating system to respectively execute on the first and second computers.

5 22. The computer program product of claim 21, wherein a master schedule includes a first process identifier identifying the first process and a second process identifier identifying the second process, the first and second process identifiers being linked together to define an executing sequence of the first and second processes, the computer program product further comprising a
10 second computer readable program code means for causing the processor to schedule the first and second processes to execute on the respective first and second computers according to the defined executing sequence.

15 23. The computer program product of claim 22, wherein the master schedule includes one or more conditional inter-relationships between the first and second processes, the computer program product further comprising a third computer readable program code means for causing the processor to schedule the first and second processes to execute based on the one or more conditional inter-relationships.

20 24. The computer program product of claim 23, wherein the one or more conditional inter-relationships include a success criteria associated with the first process, the computer program product further comprising:

fourth computer readable program code means for causing the processor to execute the first process;

25 fifth computer readable program code means for causing the processor to compare the success criteria to execution results produced by the first process; and

sixth computer readable program code means for causing the processor to determine whether the first process executed successfully based on a result of the compare.

25. The computer program product of claim 24, wherein the master table includes a third process identifier identifying a third process, the computer program product further comprising:

seventh computer readable program code means for causing the processor to execute the second process when the first process executed successfully; and

eighth computer readable program code means for causing the processor to execute the third process but not the second process when the first process did not execute successfully.

26. The computer program product of claim 25, further comprising:

ninth computer readable program code means for causing the processor to send a command to the first computer to initiate execution of the first process on the first computer;

tenth computer readable program code means for causing the processor to receive a result message from the first computer, the result message including the execution results produced by the first process; and

eleventh computer readable program code means for causing the processor to send a command to the second computer to initiate execution of the second process on the second computer when it is determined that the first process executed successfully.